시공간적 상관성을 이용한 국소 다중 탐색기반 고속 블록정합 움직임 추정

조영창[†] · 남혜영[††] · 홍성용[†††] · 이태홍[††††]

요 약

블록정합에 기초한 고속 움직임 추정 알고리듬은 탐색점의 수를 줄이기 위해 정해진 탐색패턴을 사용하며, 평균절대오차 공간에서 오차는 전역 최소해(global minimum)에 근접할수록 단조 감소한다는 가정을 바탕으로 하고 있다. 따라서, 탐색영역 내에 여러 최소점이 있는 다중 모달(multimodal) 해공간에서는 국소 최소해(local minima)에 고립될 가능성이 크며, 전역 최소해를 얻는 것은 초기 탐색점에 크게 의존한다. 이러한 현상은 서로 다른 여러 움직임이 공존하는 움직임 경계에서 더욱 부각된다. 이러한 문제점을 개선하기 위하여 본 논문에서는 시공간적으로 인접한 블록의 움직임 정보에 기초하여 탐색영역 내에 탐색 후보영역들을 정의하고, 국소 최소해로의 고립 가능성을 줄이기 위해 여러 후보영역들에 대한 다중 국소 탐색법(multiple local search method : MLSM)을 제안한다. 또한, 다중 국소 탐색법에서는 전체 후보영역들의 탐색으로 인한 부가적인 계산량을 줄이기 위해 탐색점 맵 상에 후보영역들을 표시하고 후보영역에 대한 중복탐색을 배제한다. 모의실험 결과 제안한 방법은 다른 경사법에 의한 결과보다 특히, 움직임 경계에서의 탐색에서 우수한 결과를 보였으며, PSNR에 대해서는 탐색점의 수를 증가시키지 않는 범위 내에서 전역 탐색법(full search : FS)에 의한 결과와 비슷한 결과를 얻을 수 있었다.

Fast Block Matching Motion Estimation based on Multiple Local Search Using Spatial Temporal Correlation

Youngchang Cho[†], Hyeyoung Nam[††], Sungyong Hong[†††] and Taihong Yi[††††]

ABSTRACT

Block based fast motion estimation algorithms use the fixed search pattern to reduce the search point, and are based on the assumption that the error in the mean absolute error space monotonically decreases to the global minimum. Therefore, in case of many local minima in a search region we are likely to find local minima instead of the global minimum and highly rely on the initial search points. This situation is evident in the motion boundary. In this paper we define the candidate regions within the search region using the motion information of the neighbor blocks and we propose the multiple local search method (MLSM) which search for the solution throughout the candidate regions to reduce the possibility of isolation to the local minima. In the MLSM we mark the candidate region in the search point map and we avoid to search the candidate regions already visited to reduce the calculation. In the simulation results the proposed method shows more excellent results than that of other gradient based method especially in the search of motion boundary. Especially, in PSNR the proposed method obtains similar estimate accuracy with the significant reduction of search points to that of full search.

[†] 정회원, 경운대학교 정보통신공학과 교수
[††] 영남대학교 대학원 컴퓨터공학과 박사과정
[†††] 대구과학대학 컴퓨터공학과 교수
[††††] 정회원, 영남대학교 전기전자공학부 교수

1. 서 론

움직임 추정은 움직임 보상을 이용한 필터링이나 영상압축 등, 여러 디지털 비디오 처리분야에서 중요한 역할을 하고 있으며, 영상정보의 시간적인 의존성을 분석하는데 사용되고 있다. 디지털 비디오 신호전송에서 프레임간의 중복성을 줄이기 위한 움직임 추정법으로 블록정합법은 알고리듬의 단순성과 하드웨어적인 접근의 용이성으로 인해 MPEG 계열[1,2]이나 H.261[3], H.263[4]등의 표준안 등에서 채택되어 많이 사용되고 있다. 대표적인 블록정합법으로써 전역탐색법은 탐색영역 내에서 최적의 움직임 벡터를 찾아내지만 탐색영역 전체를 찾아야 하는 많은 계산량으로 인해 다른 고속 알고리듬들이 제안되었다. 대부분의 고속 알고리듬들은 탐색점의 수를 줄임으로써 계산량을 감소하도록 고안되었으며, 대표적인 것으로 3-단계 탐색법(3SS)[5], 2D-LOG 탐색법[6], 4-단계 탐색법(4SS)[7], 블록기반 경사 하강탐색(block-based gradient descent search : BBGDS)[8] 등이 있다. 이러한 고속 알고리듬은 비교적 정확한 움직임을 추정하면서 계산량의 감소를 가져오지만, 평균 절대 오차(mean absolute error : MAE) 공간에서 오차가 전역 최소해에 근접할수록 단조 감소한다는 가정을 바탕으로 하고 있어서 실제의 경우처럼 탐색영역 내에 여러 최소점이 있는 경우, 즉, 다중 모달 해공간에서는 국소 최소점에 고립될 가능성이 크다[9]. 따라서, 전역 최소해를 찾는 것은 초기 탐색점에 크게 의존한다. 그러므로 전역 최소해를 찾기 위해서는 탐색영역 내의 여러 국소 영역에 대해 다중적으로 탐색할 필요가 있다.

이를 위해 본 논문에서는 다중 모달 해공간을 갖는 탐색영역 내에 여러 국소 해공간을 탐색 후보영역으로 재정의하고, 기존의 고속 알고리듬이 가지는 국소 최소해로의 고립문제를 개선하기 위하여 탐색영역 내에 정의된 전체 후보영역들에 대해 다중적으로 움직임 벡터를 탐색하는 다중 국소 탐색법을 제안한다. 그리고, 본 논문에서는 움직임 벡터의 시공간적인 상관성에 기초하여 이웃 블록의 움직임 벡터가 속해 있는 국소 해공간만을 후보영역으로 정의하며, 정의된 각 후보영역 내의 모든 탐색점에서는 국소 최소해로 단조 수렴하는 것으로 가정한다. 또한, 다중 국소 탐색법에서는 탐색점 맵을 사용하여 각 후보영역을 표시하고 방문한 후보영역 내의 다른 탐색점들을 배제시킴으로써 부가적인 계산량을 줄인다. 제안한 방법에서 정의한 후보영역은 실제의 국소 해공간과 정확히 일치하지는 않으나, 후보영역들로부터 얻어진 여러 국소 최소해들 가운데서 최상의 해를 얻음으로써 임의의 국소해에 대한 고립문제를 개선시킴과 동시에 전역 최소해에 근접하는 해를 얻을 수 있다. 본 논문은 2절에서 기존의 고속탐색 알고리듬의 움직임 벡터 추정방법 및 문제점을 지적하고, 3절에서는 후보영역의 정의, 탐색점 맵, 그리고 다중 국소 탐색에 의한 움직임 추정방법에 대해 다루며, 4절에서는 각 추정방법에 대한 모의실험 결과를 제시하고, 마지막으로 5절에서 결론을 맺는다.

2. 블록기반 움직임 추정 알고리듬

2.1 고속 알고리듬의 움직임 추정과정

블록정합법에서의 움직임 추정은 이전 프레임의 탐색영역 내에서 현재 블록과 가장 유사한 블록의 위치를 찾아 이를 현재 블록의 움직임 벡터로 결정한다. 이 때, 평균 절대 오차는 블록간의 유사도를 평가하기 위해 많이 사용된다. $f_t(m, n)$와 $f_{t-1}(m, n)$를 t번째 프레임과 t-1번째 프레임에서의 각각 좌·상단 좌표가 (m, n)인 블록의 밝기정보라 하고, 프레임의 크기를 M×N, 움직임 추정을 위한 블록의 크기를 K×K, 수직 및 수평방향의 움직임 벡터를 각각 u, v라 할 때, (m, n)번째 블록의 움직임 벡터 (u, v)에 대한 평균 절대 오차는 다음 식과 같이 정의된다.

$$MAE_{m,n}(u, v) = \frac{1}{K^2}\sum_{i=0}^{K-1}\sum_{j=0}^{K-1}|f_t(m+i, n+j) - f_{t-1}(m+i+u, n+j+v)| \quad (1)$$

그림 1은 대표적인 고속 움직임 추정법으로써 3SS, 2D-LOG, 4SS, BBGDS에 대한 움직임 추정과정을 표시한 것이다[10]. 각 방법들은 단계별 탐색패턴에 따라 구분된다. 그리고, 각 추정과정은 그림과 같이 탐색영역의 중심에 위치한 각 탐색패턴으로 시작하며, 각 단계에서 식(1)의 움직임 보상오차가 최소인 탐색점을 중심으로 다음 단계의 탐색패턴을 구성한다. 이러한 탐색과정은 전체 탐색영역에 대한 움직임

보상오차공간에서 오차가 전역 최소해에 근접할수록 단조 감소할 경우에는 전역 최소해를 찾을 수 있지만, 오차공간이 여러 국소 해공간으로 구성될 경우에는 초기 탐색패턴에서 최소 오차의 탐색점이 속한 국소 해공간에 고립될 가능성이 커진다.

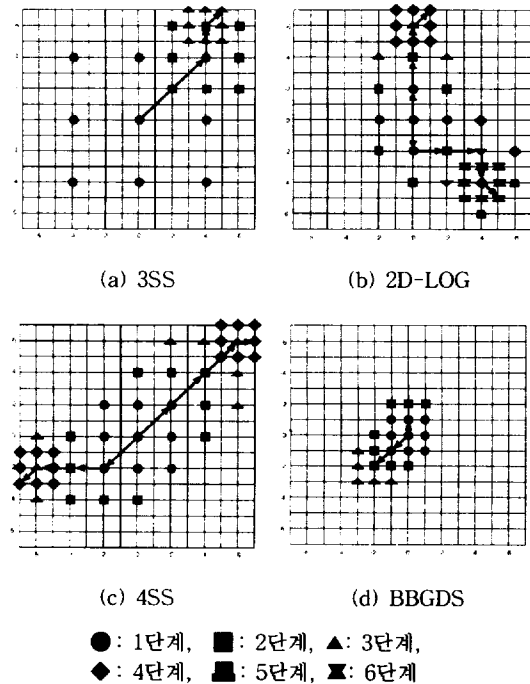



그림 1. 고속 알고리듬의 움직임 추정과정

2.2 다중 국소 탐색의 필요성

그림 2는 움직임 추정에서 탐색영역에 대한 식(1)의 평균 절대 오차 공간을 3차원으로 표시한 것이다. 이와 같이 실제의 움직임 추정에서는 탐색영역에 대한 식(1)의 평균 절대 오차 공간이 여러 국소 해공간을 갖는 것으로 나타난다. 그림 3은 그림 2를 평균 절대 오차의 크기에 따라 윤곽선으로 표시한 것으로써 여러 국소 해공간을 쉽게 보여준다.

이러한 형태의 평균 절대 오차공간은 움직임 경계에서와 같이 서로 다른 움직임을 갖는 물체들이 공존하거나 텍스쳐 패턴을 갖는 블록에 있어 더욱 두드러지게 나타난다. 따라서, 3SS, 4SS, BBGDS와 같은 단일 모달(unimodal) 해공간에 적합하도록 고안된 고속 알고리듬들은 그림 2와 같은 오차분포에 있어서 국소 최소해에 고립되기 쉽다. 그러므로, 본 논문

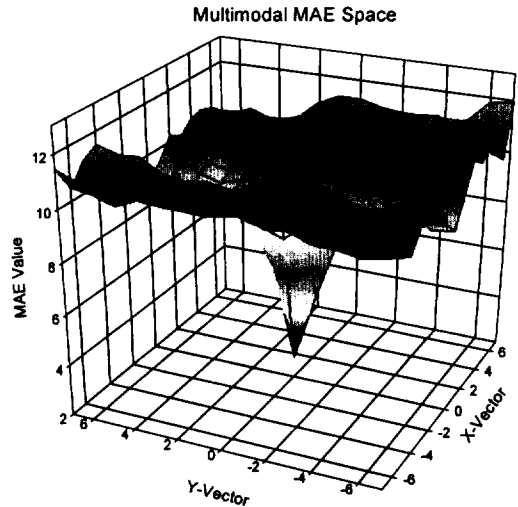



그림 2. 탐색영역에서의 MAE 분포

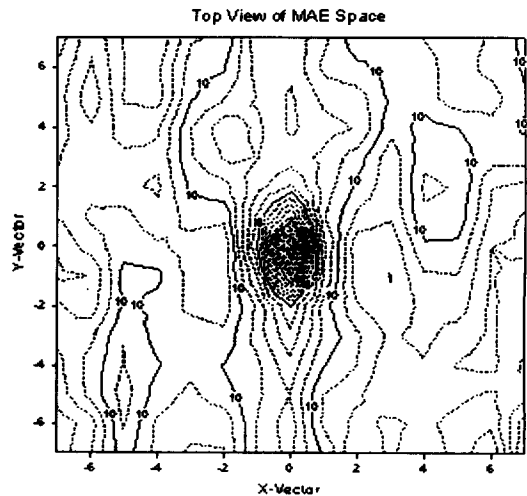



그림 3. MAE의 크기에 따른 윤곽선도

에서는 이러한 문제를 개선하기 위해 후보영역을 사용하여 국소 해공간을 재정의한 후, 후보영역 전체에 걸쳐 최소해를 찾음으로써 전역 최소해에 근접하는 해를 찾을 수 있는 다중 국소 탐색법을 제안한다.

3. 다중 국소 탐색 알고리듬

3.1 후보영역의 정의

전역 최소해를 찾기 위해서는 여러 국소 해공간에서의 국소 최소해를 찾아야 한다. 본 논문에서 제안

한 후보영역은 이러한 국소 최소해를 찾기 위해 국소 해공간을 재정의할 목적으로 사용한다. 일반적으로 블록의 움직임은 시공간적으로 인접한 이웃블록들의 움직임과 유사하다[11]. 이러한 특성은 블록이 고립된 경우를 제외하고는 여러 움직임이 공존하는 움직임 경계영역에서도 똑같이 나타난다. 아래의 식(2)는 움직임 벡터의 상관성을 나타내기 위한 식이다. 프레임 내의 전체 블록 가운데서 (m, n)번째 블록의 움직임 벡터를 $d_{(m,n)}=(u, v)$라 하고, 한 프레임 내의 (x, y)번째 블록의 움직임 벡터를 $d_{(x,y)}=(u', v')$라 할 때, 블록간의 움직임 벡터 유사도는 다음 식과 같이 정의된다.

$$S_{(m,n)}(x, y) = \frac{D^{max}_{(m,n)} - D(d_{(m,n)}, d_{(x,y)})}{D^{max}_{(m,n)}} \quad (2)$$

식(2)에서 $D^{max}_{(m,n)}$는 정규화를 위한 것으로써 다음 식과 같이 주어진다.

$$D^{max}_{(m,n)} = \max_D[D(d_{(m,n)}, d_{(i,j)}), \forall(i, j)] \quad (3)$$

식(3)에서 (i, j)는 프레임 내의 모든 블록을 가리킨다.

그림 4는 'Garden' 시퀀스에서 움직임 경계 부근의 블록들에 대해 전역 탐색법으로 추정한 움직임 벡터를 일부 표시한 것이며, 그림 5는 그림 4의 짙게 표시된 블록에 대해 식(2)의 움직임 벡터 유사도를 나타낸 것이다.

그림 4와 그림 5처럼 블록의 움직임 벡터는 이웃하는 블록의 움직임 벡터와 높은 상관성을 갖기 때문

(-1,5)	(-1,5)	(-1,5)	(-1,5)	(-1,5)	(-1,5)	(0,4)	(0,4)
(-1,4)	(-1,4)	(-1,5)	(-1,5)	(-1,5)	(-1,6)	(0,1)	(0,1)
(-1,7)	(0,1)	(-1,5)	(-1,6)	(-1,6)	(-1,6)	(0,1)	(0,1)
(0,1)	(0,1)	(-1,5)	(-1,6)	(-1,6)	(-1,6)	(0,1)	(0,1)
(0,1)	(0,1)	(-6,5)	(-1,6)	(-1,6)	(1,6)	(0,1)	(0,1)
(0,1)	(0,1)	(0,1)		(-1,6)	(-1,6)	(0,1)	(0,1)
(0,1)	(0,1)	(0,1)	(-1,6)	(-1,6)	(0,1)	(0,1)	(0,1)
(0,1)	(0,1)	(-1,5)	(-1,6)	(-1,6)	(0,1)	(0,1)	(0,1)
(0,1)	(0,1)	(-1,6)	(-1,6)	(-1,6)	(0,1)	(0,1)	(0,1)
(0,1)	(0,1)	(-1,6)	(-1,6)	(-1,6)	(0,2)	(0,2)	(0,2)
(0,2)	(0,2)	(0,6)	(0,6)	(0,6)	(0,2)	(0,2)	(0,2)
(0,2)	(0,2)	(0,6)	(0,6)	(0,6)	(0,2)	(0,2)	(0,2)
(0,2)	(0,2)	(0,6)	(0,6)	(0,6)	(0,2)	(0,3)	(0,3)

그림 4. 움직임 벡터 분포

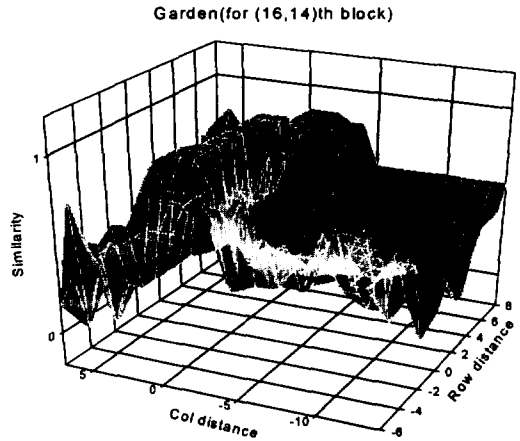



그림 5. 움직임 벡터 유사도 분포

에 현재 블록의 움직임 벡터는 이웃 블록들의 움직임 벡터가 속해있는 국소 해공간에 위치할 가능성이 크다. 따라서, 본 논문에서는 이러한 움직임 벡터의 상관성에 근거하여 이웃 블록의 움직임 벡터가 속해있는 국소 해공간만을 후보영역으로 정의한다. 이를 위해 각 후보영역은 이웃 블록의 움직임 벡터와 이 벡터가 속한 국소 해공간의 국소 최소해로부터 정해진다. k번째 이웃 블록이 속한 국소 해공간은 k번째 이웃 블록의 움직임 벡터 $d^k_i = (u^k_i, v^k_i)$와 이 벡터가 속한 국소 해공간의 국소 최소해 $d^k_c = (u^k_c, v^k_c)$로부터 식(4)의 후보영역 R^k_c로 정의된다.

$$R^k_c = \{d : D(d, d^k_c) \le D(d^k_i, d^k_c);\ \forall d = (u, v) \in R_s\} \quad (4)$$

식(4)에서 R_s는 탐색영역을 나타내며, 다음과 같이 주어진다.

$$R_s = \{(u, v) : -L \le u, v \le L\} \quad (5)$$

여기서, L은 움직임 벡터성분의 최대 크기를 나타내며, 일반적으로 L=7로 지정된다. D는 움직임 벡터간의 거리를 나타내며, 두 벡터를 각각 $d_1 = (u_1, v_1)$, $d_2 = (u_2, v_2)$라 할 때, D는 다음과 같이 정의된다.

$$D(d_1, d_2) = [(u_1 - u_2)^2 + (v_1 - v_2)^2]^{1/2} \quad (6)$$

식(4)에서 정의한 각 후보영역은 단일 모달 해공간이며, 해공간 내의 모든 점은 국소 최소해로 단조 수렴한다고 가정한다. 표 1은 본 논문에서 정의한 후보영역이 단일 모달 해공간으로 사용될 수 있음을

표. 1 후보영역 내의 탐색점에 대한 국소 최소해로의 수렴율

sequence	N_{avg}(개)	N_c(개)	N_{nc}(개)			r_e(%)
Garden	28.97	19.97	9.00	n_1	3.36	0.79%
				n_2	4.69	
				n_3	0.96	
				n_4	0.23	
Claire	21.99	18.92	3.07	n_1	1.79	0.15%
				n_2	1.12	
				n_3	0.16	
				n_4	0.03	
Tennis	43.08	21.60	21.48	n_1	8.74	1.64%
				n_2	9.75	
				n_3	2.99	
				n_4	0.71	
Salesman	23.96	17.89	6.07	n_1	2.77	2.88%
				n_2	2.32	
				n_3	0.97	
				n_4	0.69	

N_{avg} : 전체 후보영역 내의 평균 탐색점 수

N_c : N_{avg} 중에서 탐색점이 속한 후보영역 내의 국소 최소해로 수렴하는 탐색점 수

N_{nc} : N_{avg} 중에서 탐색점이 속한 후보영역 내의 국소 최소해로 수렴하지 않는 탐색점 수

n_1 : N_{nc} 중에서 후보영역의 국소 최소해보다 움직임 보상 오차가 큰 탐색점 수

n_2 : N_{nc} 중에서 다른 후보영역의 국소 최소해로 수렴하는 탐색점 수

n_3 : N_{nc} 중에서 다른 후보영역의 국소 최소해로 수렴하지 않는 탐색점 수

n_4 : n_3 중에서 후보영역의 국소 최소해보다 움직임 보상오차가 작은 탐색점 수

보이기 위해 후보영역 내의 탐색점 가운데서 정의된 후보영역 밖의 다른 해로 수렴하여 잘못된 해를 찾을 오류율 r_e에 대한 실험결과를 보여준다. 표에서 오류율 r_e는 다음과 같이 정의된다.

$$r_e = \frac{n_4}{N_{avg}} \times 100\ (\%) \tag{7}$$

식(7)에서 N_{avg}는 후보영역 내의 평균 탐색점의 개수이며, n_4는 후보영역 내의 탐색점 가운데서 후보영역 내의 국소 최소해가 아닌 다른 임의의 해로 수렴하는 탐색점의 개수이다. 각 시퀀스에 대한 실험결과, 평균 오류율은 거의 3%미만으로 나타났으며, 정의된 후보영역이 이웃 블록의 움직임 벡터가 속한 국소 해공간으로 사용되어 국소 최소해를 찾는데 이용 가능함을 보여준다.

3.2 후보영역에 대한 초기 탐색점의 결정

후보영역은 이웃 블록의 움직임 벡터가 속해 있는 국소 해공간을 나타내기 때문에 각 후보영역 내의 국소 최소해는 이웃 블록의 움직임 벡터로부터 얻어질 수 있다. 본 논문에서 사용된 이웃 블록은 그림 6과 같이 인과적인 순서와 시공간적 상관성을 고려해 정하였다. 그림 6에서 각 블록에 표시된 번호는 후보영역의 탐색에서 참조순서를 나타낸다.

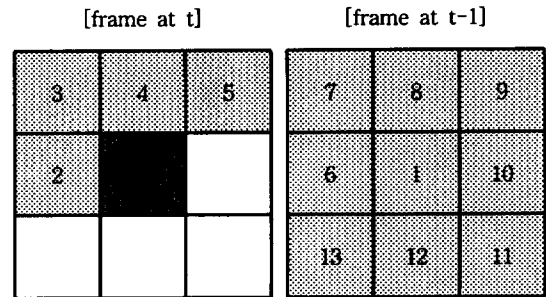



그림 6. 이웃블록의 움직임 벡터 참조순서(■: 현재 블록)

3.3 탐색점 맵(Search Point Map : SPM)의 결정

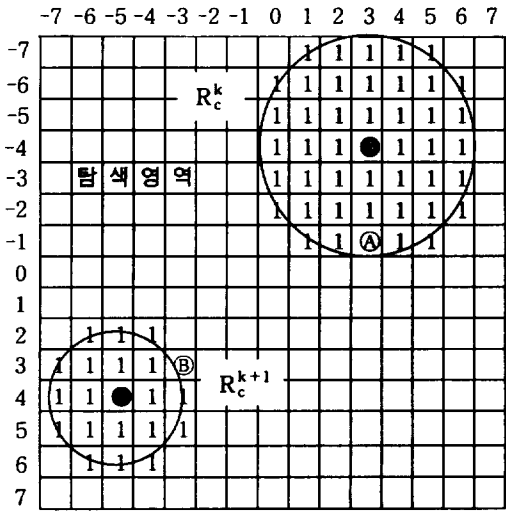
탐색점 맵은 탐색영역 내에 정의된 후보영역들을 표시하기 위해 사용되며, 각 후보영역의 탐색이 끝나면, 영역 내의 모든 탐색점들 즉, 움직임 벡터들은 탐색점 맵에서 '1'로 표시된다. 다음 식은 탐색점 맵에 표시된 k번째 이웃 블록의 움직임 벡터가 속한 후보영역 R_c^k를 나타낸다.

$$S(u,v) = 1,\ \text{if}\ (u,v) \in R_c^k\ \text{and}\ (u,v) \in R_s \tag{8}$$

식(8)에서 S는 탐색점 맵을 나타낸다.

그림 7은 탐색영역 내에 표시된 두 후보영역 R_c^k, R_c^{k+1}을 표시한 그림이다. 탐색점 맵으로부터 같은 후보영역 내의 탐색되지 않은 다른 움직임 벡터들도 같은 국소 최소해로 수렴할 것이기 때문에 이후의 탐색에서는 제외된다. 그러므로, 탐색영역 내의 움직임 추정이 후보영역 단위로 행해지게 되고, 화소방문 횟수를 추가로 증가시키지 않으면서 추정된 움직임 벡터의 정확성을 증대시킬 수 있다.

3.4 다중 국소 탐색에 의한 움직임 추정과정

제안한 다중 국소 탐색법은 이웃블록의 움직임 벡

● : $d_c^k = (u_c^k,\ v_c^k)$ ● : $d_c^{k+1} = (u_c^{k+1},\ v_c^{k+1})$

Ⓐ : $d_i^k = (u_i^k,\ v_i^k)$ Ⓑ : $d_i^{k+1} = (u_i^{k+1},\ v_i^{k+1})$

그림 7. 두 후보영역에 대한 탐색점 맵

터를 이용하여 전체 후보영역들에 걸쳐 전역 최소해를 찾는다. 또한, 탐색점 맵에 표시된 동일 후보영역을 중복해서 탐색하지 않음으로써 다중 탐색에 의한 계산량을 절감시킨다. 제안한 방법에 의한 움직임 추정절차는 다음과 같다.

(i) 프레임에서 (m, n)번째 블록에 대한 움직임 벡터를 추정한다. k=1로 설정한다. 여기서, k는 이웃 블록의 참조순서를 결정하며, 참조순서는 그림 6과 같다.

(ii) (m, n)번째 블록에 대한 탐색점 맵을 초기화한다.

$$S(u,v) = 0,\quad \forall (u,v) \in R_s$$

(iii) k번째 이웃블록의 움직임 벡터 $d_i^k = (u_i^k,\ v_i^k)$에 대해 $S(u_i^k,\ v_i^k) = 1$이면, (v)과정으로 이동하고, $S(u_i^k,\ v_i^k) = 0$이면, $d_i^k = (u_i^k,\ v_i^k)$를 초기 탐색 시작점으로 하여 BBGDS 탐색에 의해 국소 최소해 $d_c^k = (u_c^k,\ v_c^k)$를 구한다. 그리고, 국소 최소해와 이에 대한 평균절대오차를 저장한다.

(iv) d_i^k와 d_c^k로부터 R_c^k를 구하고, 이를 탐색점 맵에 표시한다.

(v) k=k+1로 설정한다.

(vi) $k \le N_k$이면, (iii)과정으로 이동하여 다음 이웃 블록에 대한 후보영역 탐색을 반복한다. $k > N_k$이면, (vii)과정으로 이동한다. 여기서, N_k는 그림 6의 순서에 따라 참조되는 이웃 블록의 수를 나타내며, 본 논문에서는 N_k=13으로 설정한다.

(vii) 다음 블록에 대한 움직임 벡터 추정을 위해 (i)과정에서 (vi)과정까지 반복한다.

(i)~(vi)과정동안 평균절대오차가 최소인 국소 최소해 $d_c^k = (u_c^k,\ v_c^k)$를 전역 최소해로 결정한다. 그림 8은 다중 국소 탐색에 대한 흐름도를 나타낸 것이다.

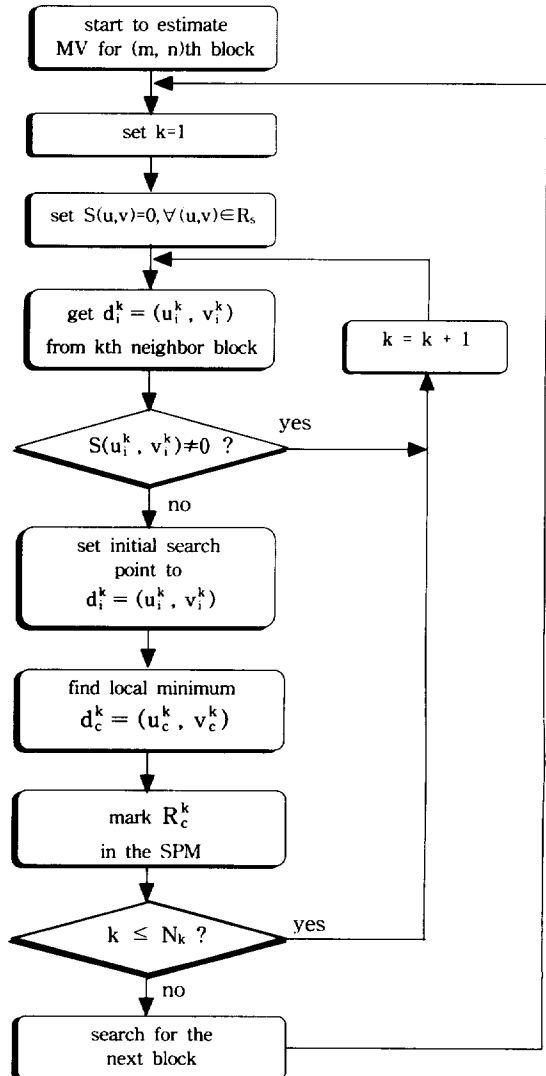



그림 8. 다중 국소 탐색에 대한 흐름도

4. 실험결과

본 논문에서 제안한 움직임 추정법을 검증하기 위해 모의실험을 실시하였다. 실험에서 사용된 영상은 Garden(352×240 : 100 프레임), Claire(352×240 : 100 프레임), Salesman(352×240 : 100 프레임) 그리고 Tabel Tennis(360×240 : 100 프레임)이며, 프레임별로 움직임 보상영상에 대한 PSNR과 방문 화소수(NSP)에 대해서 제안한 방법과 기존의 방법을 비교하여 평가하였으며, 각 방법에 대한 움직임 벡터의 니들그램(needlegram)을 통해 전역 탐색법에 의한 결과와 비교하였다. 식(9)과 식(10)은 PSNR 계산을 위해 사용하였다.

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \tag{9}$$

$$MSE = \frac{1}{N\times M}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}[f_t(i,j) - \hat{f}_t(i,j)]^2 \tag{10}$$

식(10)에서 $f_t(i,j)$는 t번째 프레임 영상에서 (i, j)번째 화소값을 나타내며, $\hat{f}_t(i,j)$는 움직임 보상된 t번째 프레임 영상의 (i, j)번째 화소값을 나타낸다.

그림 9은 'Garden' 시퀀스에 대해 각 방법에 의한 PSNR을 프레임별로 나타낸 것이다. 제안한 추정법에 의한 결과, PSNR은 전역 탐색법에 의한 결과와 비슷하게 나타났으며, 기존의 방법인 3SS, 4SS, BBGDS에 의한 결과보다 우수한 것으로 확인되었다. 그림 10은 각 방법에 의한 탐색에서 프레임에 따라 평균 방문화소수를 표시한 것이다. 제안한 방법은

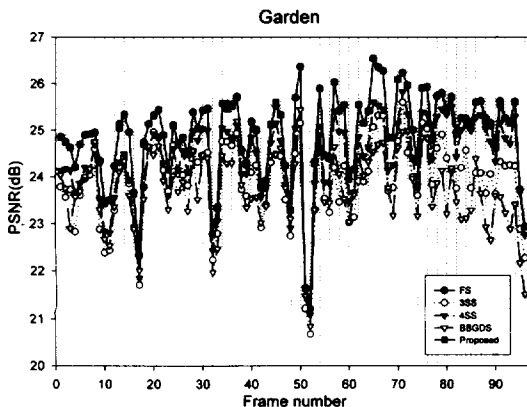



그림 9. 'Garden' 영상의 전체 프레임에 대한 각 방법별 PSNR 비교

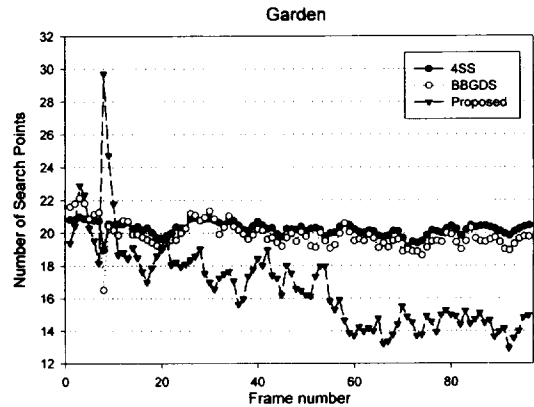



그림 10. 'Garden' 영상의 전체 프레임에 대한 각 방법별 NSP 비교

10번째 프레임 근처에서 다소 많은 화소를 방문하지만, 전반적으로 BBGDS나 4SS보다 방문화소가 작게 나타났다. 이는 이웃블록에 의한 탐색과 후보영역에 대한 중복탐색을 배제함으로써 얻어지는 결과이다. 이 밖의 'Claire', 'Table tennis' 및 'Salesman' 시퀀스에 대한 실험결과를 PSNR과 NSP에 대해 표 2와 표 3에 표기하였다. 제안한 방법에 의한 NSP는 'Claire'와 'Table tennis'의 경우, 약간 많게 나타났지만, 'Garden'과 'Salesman'의 경우, BBGDS보다 작게 나타났다. 또한, 표 2의 PSNR에 대해서는 'Claire'와 'Salesman'의 경우, 많은 움직임이 없기 때문에 비교적 작은 차이를 보였지만, 'Garden'과 'Table tennis'

표 2. 영상별 각 방법에 대한 평균 PSNR 비교(dB)

Average PSNR for Methods					
Methods	FS	3SS	4SS	BBGDS	MLSM
Garden	24.89	23.92	24.39	23.71	24.88
Claire	23.03	23.03	23.03	23.03	23.03
Tennis	23.26	22.53	22.94	22.58	23.22
Salesman	27.12	27.09	27.10	27.10	27.11

표 3. 영상별 각 방법에 대한 평균방문횟수(NSP) 비교

Average # of Search Points per block					
Methods	FS	3SS	4SS	BBGDS	MLSM
Garden	225	25	20.26	19.84	16.71
Claire	225	25	17.11	9.59	10.49
Tennis	255	25	20.32	17.17	22.24
Salesman	225	25	17.39	11.38	11.31

의 경우, 제안한 방법에 의한 결과가 훨씬 양호하게 나타났다. 이러한 차이는 그림 11의 움직임 벡터에 대한 니들그램을 통해서도 분명히 확인할 수 있다. 그림 11의 'Garden' 시퀀스에 대한 경우, 3SS이나 4SS에 의한 추정결과, 배경 및 움직임 경계에서의 움직임 벡터가 잘못 추정된 것을 볼 수 있으며, 특히, 움직임 경계에서는 BBGDS에 의한 결과가 가장 저조한 것으로 나타났다. 반면, 제안한 방법에 의한 추정결과, 움직임 경계가 있는 나무윤곽 근처에 위치한 블록의 움직임이 정확히 추정된 것을 볼 수 있으며, 움직임 경계에서나 배경 그리고 물체 내에서 제안한 방법에 의한 추정결과가 전역 탐색법에 의한 결과에 가장 근접함을 볼 수 있다. 그러므로, 서로 다른 여러 움직임의 공존으로 인한 다중 모달 해공간에서는 여러 후보영역에 걸쳐 해를 찾는 방법이 효과적임을 알 수 있다.

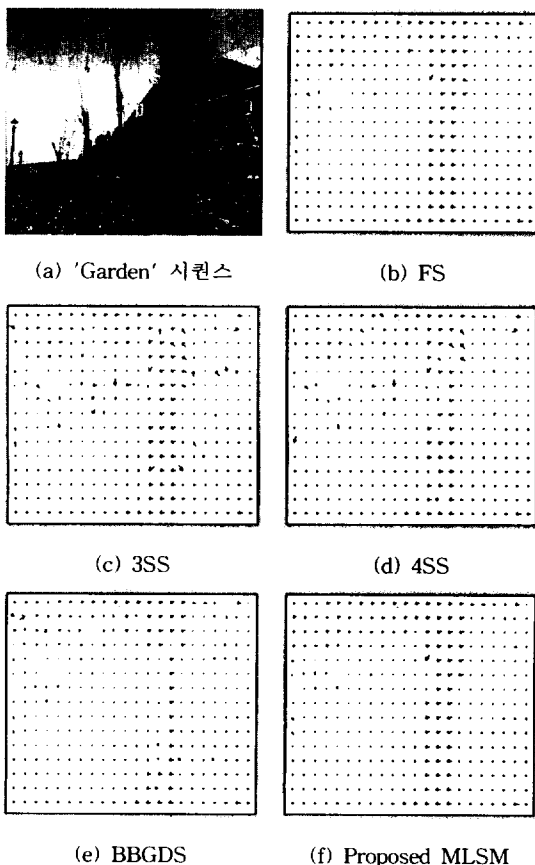



그림 11. 'Garden' 영상에 대한 각 방법별 움직임 벡터의 니들그램

5. 결 론

본 논문은 탐색영역 내에서 여러 후보영역들을 정의하고 전체 후보영역들에 대해 다중적으로 움직임을 추정하여 탐색영역 내에서 전역 최소해에 근접하는 해를 찾는 다중 국소 탐색법을 제안하였다. 제안한 다중 국소 탐색법은 각 후보영역을 탐색점 맵에 표시하고 같은 후보영역에 대한 중복탐색을 배제함으로써 정의된 후보영역 단위로 해를 찾기 때문에 전체 후보영역에 대한 부가적인 계산량을 절감할 수 있다. 제안한 방법에 대한 검증결과 특히, 움직임 경계영역에서 정확한 움직임을 추정할 수 있음을 확인할 수 있었으며, 계산량의 큰 증가없이 움직임 추정의 정확성을 개선시킬 수 있었다.

참 고 문 헌

[1] ISO/IEC 11172-2(MPEG-1 Video), "Information Technology-Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbits/s: Video," 1993.

[2] ISO/IEC 13818-2 | ITU-T H.262((MPEG-2 Video), "Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video," 1995.

[3] ITU-T Recommendation H.261, "Video codec for audiovisual services at P×64 k bits," *ITU-T*, March 1993.

[4] ITU-T DRAFT H.263, "Video coding for Low bitrate communication ITU-T," December 1995.

[5] Her-Ming Jong, Liang-Gee Chen, and Tzi-Dar Chiueh, "Accuracy Improvement and Cost Reduction of 3-Step Search Block Matching Algorithm for Video Coding", *IEEE Trans. Circuits Syst. Video Tech.*, vol. 4, no. 1, pp. 88-90, Feb. 1994.

[6] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun., vol. COM-29*, pp. 1799-1808, Dec. 1981.

[7] L. M. Po and W. C. Ma, "A novel four-step

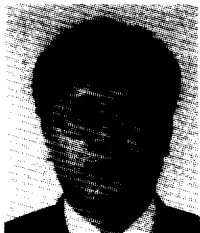
search algorithm for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 6, no. 3, pp. 313-317, Jun. 1996.

[8] Lurng-Kuo Liu and Ephraim Feig, "A Block-Based Gradient Descent Search Algorithm for Block Motion Estimation in Video Coding," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 6, no. 4, pp. 419-422, Aug. 1996.

[9] Man F. So and Angus Wu, "Four-step Genetic Search For Block Motion Estimation," *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'98*, vol. 3, May 1998.

[10] C. K. Cheung, "Fast motion estimation techniques for video compression," M. Phil. Dissertation, CityU Image Processing Lab, Department of Electronic Engineering, City University of Hong Kong, July 1998.

[11] Jyi-Chang Tsai, Chaur-Heh Hsieh, Shiuh-Ku Weng, and Mao-Fu Lai, "Block-matching motion estimation using corrrelation search algorithm," Signal Processing : Image Communication, 1998.

조 영 창

1994年 8月 嶺南大學校 電氣工學科(工學士)
1996年 8月 嶺南大學校 大學院 電子工學科(工學碩士)
1998年 8月 嶺南大學校 大學院 電子工學科(博士修了)
관심분야 : 움직임 추정, 동영상 복원

남 혜 영

1995年 2月 曉星女子大學校 電子計算學科(理學士)
1997年 2月 大邱曉星가톨릭大學校 大學院 電算統計學科(理學碩士)
1999年 8月 嶺南大學校 大學院 컴퓨터工學科(博士科程)
관심분야 : 컴퓨터 비젼, 패턴인식

홍 성 용

1987年 嶺南大學校 電子工學科(工學士)
1989年 嶺南大學校 大學院 電子工學科 電子通信專攻(工學碩士)
1994年 嶺南大學校 大學院 電子工學科 電子通信專攻(工學博士)
관심분야 : 움직임 추정 및 영상복원

이 태 홍

1975年 2月 서울大學校 工業教育學科 電子工學專攻 卒業
1976年 5月～1979年 10月 東洋精密(株) 中央硏究所 책임연구원
1980年 2月 서울大學校 大學院 工業教育學科 電子工學專攻 卒業(碩士)
1980年 1月～1981年 2月 金星電氣(株) 技術硏究所 技佐
1981年 3月～현재 嶺南大學校 電氣電子工學部 敎授
1996年 10月～1997年 10月 韓國通信學會 大邱・慶北 支部長
1997年 2월 서울大學校 大學院 電子工學科 卒業(博士)
관심분야 : 신호처리, 영상처리